

## Method and apparatus to manage picture and pageset for document processing

**Patent number:** DE4313959  
**Publication date:** 1993-11-04  
**Inventor:** MOTOYAMA TETSURO (US)  
**Applicant:** RICOH KK (JP)  
**Classification:**  
 - **international:** G06F3/12; G06F17/21; G06F17/22; G06F17/27;  
 G06F3/12; G06F17/21; G06F17/22; G06F17/27; (IPC1-7): G06F15/20  
 - **european:** G06F3/12T; G06F17/21F; G06F17/22T; G06F17/22T2;  
 G06F17/27  
**Application number:** DE19934313959 19930428  
**Priority number(s):** US19920876601 19920430

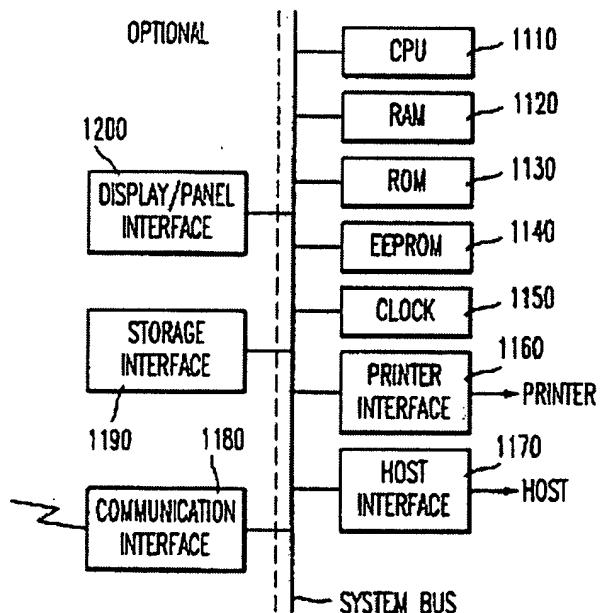
Also published as:

US5319748 (A1)  
 JP6096072 (A)

[Report a data error here](#)

Abstract not available for DE4313959  
 Abstract of corresponding document: **US5319748**

A printer/display control apparatus for processing documents represented in a structured hierarchical page description language. The documents are provided as a document data stream defined by a hierarchical structure having as a top level a "Pageset" of "Picture". Both the Pageset and the Picture consist of an optional Prologue or an optional Prologue and an optional Body. The Body of a Pageset consists of zero or more Pagesets or Pictures. The Body of Picture consists of zero or more Pictures or Tokensequence. Each Picture or Pageset has a "begin" and "end" used to define in a document data stream the beginning and end of the definitions that make up that entity. This invention provides an efficient method to process the "begin" and "end" of the document data stream.



Data supplied from the esp@cenet database - Worldwide



⑯ BUNDESREPUBLIK  
DEUTSCHLAND



DEUTSCHES  
PATENTAMT

⑯ Offenlegungsschrift  
⑯ DE 43 13 959 A 1

⑮ Int. Cl.<sup>5</sup>:  
G 06 F 15/20

DE 43 13 959 A 1

⑯ Aktenzeichen: P 43 13 959.0  
⑯ Anmeldetag: 28. 4. 93  
⑯ Offenlegungstag: 4. 11. 93

⑯ Unionspriorität: ⑯ ⑯ ⑯  
30.04.92 US 876601

⑯ Erfinder:  
Motoyama, Tetsuro, San Jose, Calif., US

⑯ Anmelder:  
Ricoh Co., Ltd., Tokio/Tokyo, JP

⑯ Vertreter:  
Schwabe, H., Dipl.-Ing.; Sandmair, K., Dipl.-Chem.  
Dr.jur. Dr.rer.nat.; Marx, L., Dipl.-Phys. Dr.rer.nat.,  
Pat.-Anwälte, 81677 München

Prüfungsantrag gem. § 44 PatG ist gestellt

⑯ Einrichtung und Verfahren zum Steuern der Darstellung einer Vorlage

⑯ In einer Drucker/Anzeige-Steuereinrichtung zum Verarbeiten von Vorlagen, welche in einer strukturierten, hierarchischen Seitenbeschreibungssprache dargestellt sind, sind die Vorlagen als ein Vorlagendatenstrom vorgesehen, welcher durch eine hierarchische Struktur festgelegt ist, welcher als einen oberen Pegel einen "Seitensatz" von einem "Bild" hat. Sowohl der Seitensatz als auch das Bild bestehen aus einem wahlweisen Prolog oder einem wahlweisen Prolog und einem wahlweisen "Körper". Der Körper eines Seitensatzes besteht aus null oder mehr Seitensätzen oder Bildern. Der "Körper" eines Bildes besteht aus null oder mehr Bildern oder Einheiten-(token)Folge jedes Bildes oder jeder Seitensatz hat einen "Beginn" und ein "Ende", die verwendet werden, um in einem Vorlagendatenstrom den Beginn und das Ende der Definitionen festzulegen, welche dieses Objekt (entity) bilden. Durch die Erfindung ist ein effizientes Verfahren geschaffen, um "den Beginn" und das "Ende" des Vorlagendatenstroms zu verarbeiten.

DE 43 13 959 A 1

## Beschreibung

Die Erfindung betrifft eine Einrichtung zum Steuern der Darstellung einer Vorlage nach dem Oberbegriff des Anspruchs 1, 2 oder 13 und betrifft ferner ein Verfahren zum Steuern der Darstellung einer Vorlage nach dem Oberbegriff des Anspruchs 15. Ferner betrifft die Erfindung generell ein rechnergesteuertes Drucken, Anzeigen oder eine andere Darstellung von Vorlagen, welche Text, graphische Komponenten und/oder Bilder haben können, und betrifft insbesondere ein Verfahren und Systeme zum Verarbeiten des Beginns und Endes von Bildern oder Seitensätzen in einem Vorlagendatenstrom sowie andere Daten, welche in dem Vorlagendatenstrom festgelegt sein können. Die Erfindung betrifft auch eine Methode, um die Spur des Bereichs oder Zustands der Darstellungsvorrichtung zu halten.

In diesem Zusammenhang schließt die Darstellung einer Vorlage auf einer Seite das Drucken auf einem festen Medium, wie Papier, einem transparenten Material u.ä. ein Darstellen der Seite auf einer visuellen Anzeige oder ein Übertragen der Vorlage in einer darzustellenden oder zu speichernden Form an eine andere Einrichtung mit ein. Die vorliegende Anmeldung hat auch Bezug zu einer anhängigen, am 17. Oktober 1991 eingereichten US-Patentanmeldung, S.N. 07/778 578 auf deren Inhalt hiermit Bezug genommen wird. Die vorliegende Erfindung hat ferner Bezug zu einer gerade eingereichten US-Patentanmeldung mit dem Titel "Verfahren und Einrichtung zum Behandeln einer Implikation von externen Verweisen in einer Vorlagenverarbeitungssprache", auf welche hiermit ebenfalls Bezug genommen wird.

Durch die Entwicklung von Laserdruckern in den frühen Siebziger-Jahren wurde eine Möglichkeit für hochqualitatives und preiswertes Drucken von Vorlagen geschaffen, welche nicht nur gleiche Daten, sondern auch übliches graphisches Material enthalten. Bei dem grundlegenden Verfahren zum Steuern dieser Art Drucker mittels Befehlen wurde eine sogenannte "Escape folge" verwendet, deren Befehle der Art Befehle ähnlich sind, welche in einer DIABLO® Befehlsfolge verwendet sind. Diese Befehle werden durch typische Zeichendaten in einem Datenstrom unterschieden, wobei jedem Befehl ein spezielles Byte, im allgemeinen ein Escape-Zeichen (ASCII 27) vorausgeht. Diese Methodologie war annehmbar für mit niedriger Geschwindigkeit arbeitende Vorrichtungen und für Vorrichtungen, welche begrenzte Mengen Graphiken drucken, wie Typenrad- oder Punktmatrixdrucker; sie war allerdings nicht zum Drucken von Vorlagen geeignet, welche beachtliche Mengen Text und grafische Materialien verknüpfen, da keine Vorkehrungen für die großen Datenmengen vorgesehen waren, welche eine Graphik benötigen kann.

Als Antwort auf die Beschränkungen im Falle der Escape-Steuерfolge wurden verschiedene Arten von "Seitenbeschreibungssprachen" (PDL) entwickelt, um Laser- oder andere Arten von Seitendruckern zu steuern. Eine rückwärts wirkende Kompatibilität war bezüglich der meisten dieser Laserdrucker vorgesehen, so daß sie in der Lage waren, herkömmliche Escape-Folge-Befehle anzunehmen, wie sie in den vorerwähnten Druckern des DIABLO® oder EPSON®-Typs verwendet werden können. Beispiele der herkömmlichen Seitenbeschreibungssprachen, welche entsprechend den Beschränkungen entwickelt wurden, die den Escape-Folgebefehlen anhaften, waren das System von Adobe Systems, Inc., und das Interpress®-System von Xerox. Es

wurden noch andere gesetzlich geschützte Seitenbeschreibungssprachen geschaffen, um die Beschränkungen zu umgehen, die bei der Verwendung von Escape-Folgebefehlen zu erwarten sind.

5 Einer der Nachteile der herkömmlichen Seitenbeschreibungssprachen, wie PostScript®, wird augenscheinlich bezüglich der Vorkehrungen, die gemacht werden, damit eine ganz bestimmte Vorlage eine neue Definition enthalten kann, wie eine neue Ressourcen-Definition (z. B. eine zusätzliche Schriftart oder Grafik, welche in der Vorlage zu verwenden ist) oder eine neue Wörterbuch-Definition. Diese neuen Definitionen können irgendwo in einem Vorlagendatenstrom erscheinen und sind schwierig zu verarbeiten und zu identifizieren. Folglich müssen der gesamte Inhalt einer Vorlage überprüft werden, um festzustellen, ob ein gegebener Drucker Ressourcen hat, die notwendig sind, um eine ganz bestimmte Vorlage zu drucken oder um die Vorlage in der gewünschten Weise zu verarbeiten. Folglich kann es vorkommen, daß das Drucken einer Vorlage an irgendeiner Stelle während des Druckprozesses sogar kurz vor dem Verarbeitungsende ausfällt, da der Drucker nicht in der Lage ist, Befehle zu befolgen, welche in der Seitenbeschreibungssprache der Vorlage enthalten sind.

20 Eine weitere Schwierigkeit bei einem PostScript®-System besteht darin, daß, um eine vorgegebene Seite einer Vorlage zu drucken, wenn es nicht die erste Seite ist, es notwendig ist, die ganze PDL-Beschreibung aller vorhergehenden Seiten der Vorlage zu lesen, um den Zustand der Vorlagenseiten-Einstellparameter (d. h. eines Ressourcen-Deklaration, eine Wörterbuch-Definition u.ä.) zu bestimmen, bevor die Verarbeitung einer ganz bestimmten Seite ausgeführt werden kann. Mit anderen Worten, eine Drucksteuereinheit oder ein Drucktreiberprogramm müssen den gesamten PDL-Vorlagendatenstrom lesen, um die Wirkung jedes Seitenstellbefehls in Betracht zu ziehen, welcher den Zustand des Druckens zwischen dem Anfang der Vorlage und der ganz bestimmten Seite ändern kann. Obwohl dieses Abtasten verhältnismäßig unkompliziert ist, erfordert es eine beträchtliche Verarbeitungszeit und ist folglich eine unwirksame Lösung des Problems. Ferner wird es kompliziert, das eine Kennzeichnung des Anfangs und Endes von Seiten und von grafischen Darstellungen in PostScript-Vorlagen nicht ganz unkompliziert ist und eine beachtliche Verarbeitungszeit erfordert. PostScript® entwickelte die Vorlagen-Strukturierungskonvention mit Hilfe von Kommentaren, um verschiedene Ressourcen und Seitenenden (breaks) zu unterscheiden. Es gibt jedoch diese Konvention und in der Beschreibungssprache gibt es nichts, um dieser Konvention Gelung zu verschaffen. Eine Verletzung der Konvention wird nicht als Fehler gekennzeichnet.

25 30 35 40 45 50 55 60 65 Die herkömmlichen Seitenbeschreibungssprachen, einschließlich PostScript und InterPress weisen verschiedene Verbesserungen bezüglich der herkömmlichen Escape-Folgen auf, wie ein Vorsehen von Werkzeugen für solche Gegenstände, wie Ressourcen-Deklarationen, Kontext-Deklarationen, Wörterbücher, die Benutzung von Speicherstapeln u.ä. Diese Sprachen ließen in einigen Fällen auch dynamische Änderungen in dem Druckerzustand zu, so beispielsweise die Möglichkeit, Schriftarten oder graphische Bilder zu den Ressourcen hinzuzufügen, die in dem Drucker verfügbar sein können. Einige dieser Merkmale sind in solchen, im allgemeinen verfügbaren Referenzarbeiten dokumentiert, wie in dem "PostScript Language Reference Ma-

"nual" von Adobe System Incorporated, und im "Post-Script Language Programm Design", beide von Addison-Wesley Publishing Company. Andere Seitenbeschreibungssprachen sind ebenfalls in verschiedenen technischen und Referenzbüchern beschrieben, wie "Interpress, The Source Book" von Harrington et.al (Simon and Schuster, Inc, 1988). Eine der Schwierigkeiten, welche vorstehend bezüglich des PostScript-Systems angeführt sind, ist die, daß eine Kennzeichnung des Beginns und des Endes eines Bildes oder einer Seite einen beträchtlichen Bearbeitungsaufwand erfordert, und diese Sprachen sind nicht in einer logischen Weise geschaffen, bei welcher die Vorbereitung der Vorlage eine beachtliche zusätzliche Verarbeitung erfordern kann, um diese Punkte zu bestimmen. Tatsächlich ist ein Verarbeiten der gesamten Vorlage, "wie wenn sie gedruckt würde" erforderlich, selbst wenn nur eine Seite oder ein Abschnitt von Seiten, wie ein Kapitel in einem langen Dokument, benötigt werden. Ferner kann das Verarbeiten nicht unkompliziert und einfach genug sein, um für preiswerte Prozessoren bei der Drucker/ Anzeigevorrichtung verwendet werden zu können. Diese Schwierigkeit erhöht die Kosten oder mindert die Flexibilität der herkömmlichen Systeme. Eine Basis für eine vorgeschlagene standardisierte Seitenbeschreibungssprache ist von verschiedenen Leuten gemacht worden, einschließlich den Erfindern der vorliegenden Anmeldung und wird gerade zu einer internationalen Norm entwickelt. Der Vorschlag liegt zum gegenwärtigen Zeitpunkt in Form eines Entwurfs einer Abteilung des 150 als ISO/IEC DIS 10180 vor und wird als "Informationsverarbeitung-Text-Übermittlung-Standard-Seitenbeschreibungssprache" bezeichnet. Der aktuelle Entwurf ist von 1991 datiert.

Gemäß der Erfindung soll daher eine Einrichtung und ein Verfahren zum Bestimmen und zum effizienten Verarbeiten des Beginns und des Endes von Bildern und/oder Seitensätzen in einem Vorlagendatenstrom geschaffen werden, um so das Verarbeiten von Vorlagendatenströmen zu beschleunigen. Ferner soll gemäß der Erfindung eine wirksame Methode und Einrichtung geschaffen werden, welche eine Spur des Materialbereichs und -zustands, die in der Definition für jeden Vorlagendatenstrom (sogenannte Prolog) festgelegt sind, in wirksamer Weise halten können. Darüber hinaus soll gemäß der Erfindung eine Einrichtung und Methode geschaffen werden, um in einem eingegebenen Vorlagendatenstrom eine vorgegebene Seite oder ein vorgegebenes Bild auszuwählen, um es in effizienter Weise zu verarbeiten, ohne daß ein Verarbeiten aller vorhergehenden Seiten der Vorlage erforderlich ist.

Dies ist gemäß der Erfindung durch ein effizientes Verarbeitungsverfahren in DIS 10 180 geschaffen. Gemäß DIS 10 180 ist jeder Vorlagendatenstrom in einer Struktur geschaffen, welche entweder ein Seitensatz oder ein Bild ist. Der Seitensatz und Bildelemente bestehen aus einem wahlweisen Prolog, welcher Definitionen und erklärende Befehle enthält, und aus einem wahlweisen Körper. Ein Seitensatz-Körper besteht aus null oder mehr Seitensätzen oder Bildern, während ein Bildkörper aus null oder mehr Bildern oder Einheiten – bzw. Tokenfolgen besteht. Die Einheitenfolge, welche spezifische Einheiten oder Befehle enthält, um spezifische Bilder zusammen mit notwendigen Operatoren festzulegen, wird Inhalt (content) genannt, während andere Elemente als Struktur bezeichnet werden. Dieses Strukturen stellen die Umgebung für den Inhalt dar, um entsprechende Ausgangsbilder zu erzeugen. Die Prolog-

wirkungen in einem Bild oder einem Seitensatz gelten bis zu Ende des Bildes oder des Seitensatzes. Folglich beeinflußt der Prolog eines Bildes in dem hierarchischen Raum nicht das Peer-Bild, während der Prolog eines Elter-Bildes oder Seitensatzes die Kind-Bilder beeinflußt. Mit der vorliegenden Erfindung ist effizient diese hierarchische Baumstruktur im Vergleich zu dem Bereich des Prologs mit Hilfe eines Stapels und verschiedener Zeiger behandelt.

Eine baumartig verbundene Hierarchie ermöglicht in vorteilhafter Weise das Verarbeiten eines Teils der Vorlage, indem dieser Teil der Vorlage und diejenigen Teile, welche in der Hierarchie höher sind, direkt adressiert werden, ohne daß eine Verarbeitung der übrigen Einzelheiten in den verschiedenen Zweigen des hierarchischen Baums notwendig sind. Mit anderen Worten, nur strukturelle Definitionen, welche in dem hierarchischen Baum vorkommen und welche oben ein vorgegebener Teil der Vorlage sind, müssen verarbeitet werden. Hierdurch wird das Leistungsvermögen der Verarbeitung der Vorlage erhöht und auch die Bestimmung der Art von Ressourcen, welche in der Druck- oder der Anzeigevorrichtung benötigt werden, vor dem Beginn des tatsächlichen Druckens der Vorlage erleichtert. Hierdurch werden die Geschwindigkeit und das Leistungsvermögen erhöht, mit welchen die verschiedenen Vorrichtungen die Vorlage drucken oder anzeigen.

Nachfolgend wird die Erfindung anhand von bevorzugten Ausführungsformen unter Bezugnahme auf die anliegenden Zeichnungen im einzelnen erläutert. Es zeigen:

Fig. 1 eine Darstellung der Grundstruktur eines Vorlagen-Datenstroms, wobei eine Vorlage entweder als ein Seitensatz oder ein Bild festgelegt ist, das in der strukturierten standardisierten Seitenbeschreibungssprachenform dieser Erfindung dargestellt ist;

Fig. 2 eine Darstellung der hierarchischen Struktur einer in Fig. 1 dargestellten Vorlage;

Fig. 3 eine Darstellung einer repräsentativen Grammatik eines Bildes und Seitensatzes gemäß der in Fig. 2 dargestellten Struktur;

Fig. 4 eine schematische Darstellung der generellen Architektur der Bild-Seitensatz-Stapelanordnung gemäß der Erfindung;

Fig. 5 eine Darstellung der Struktur einer Bild-Seitensatz-Prolog-Datenstruktur und der verbundenen Liste, welche diese Struktur begleitet;

Fig. 6 eine Darstellung der Datenstruktur der Verbindungsliste, welche eine Verbindung zu einer externen Deklaration unterstützt, welche in einem Vorlagendatenstrom enthalten sein kann;

Fig. 7 ein Flußdiagramm der Verarbeitung, welche für den Prozessor notwendig ist, um den Beginn eines Bildes zu behandeln;

Fig. 8 ein Flußdiagramm der Routines, welche für den Prozessor notwendig sind, um den Beginn eines Seitensatzes zu behandeln;

Fig. 9 ein Flußdiagramm der Verarbeitung, welche für eine Bildende-Routine notwendig ist;

Fig. 10 ein Flußdiagramm der Verarbeitung, welche für eine Seitensatzende-Routine notwendig ist;

Fig. 11 eine beispielhafte Ausführung dieses Systems entweder in einem Host-Computer, einem Druck-Server oder Drucker-Steuereinheit gemäß der Erfindung;

Fig. 12A bis 12C eine Darstellung von drei möglichen Realisierungsumgebungen, und

Fig. 13 ein erweiterter oberer Pegel-Durchbruch eines Vorlagendatenstroms, wie er in Fig. 2 dargestellt ist

und wie er gemäß der Verbindung verarbeitet worden ist.

In Fig. 1 ist ein grundlegendes Vorlagendatenstrom-Layout dargestellt, in welchem eine Vorlage als eine Gruppierung von Seitensätzen definiert sein kann. Unter Bezugnahme auf Fig. 11 wird deutlich, daß ein Ziel-System, welches in der Lage ist, den Vorlagen-Datenstrom aufzunehmen, im allgemeinen zumindest eine CPU-Einheit, einen Speicher im allgemeinen in Form eines ROM- oder eines RAM-Speichers, eine Ein-/Ausgabeeinrichtung, um den Vorlagendatenstrom aufzunehmen, und eine Einrichtung zum Anzeigen/Drucken eines sich ergebenden Ausgabebildes. Das sich ergebende Ausgabebild kann durch eine Bildrastervorrichtung erzeugt werden, welche unmittelbar mit der CPU-Einheit verbunden sein kann, oder ihre eigene klare geräte-abhängige Abbildung haben kann, welche dann natürlich eine Übersetzung eines Vorlagendatenstroms in die Art Abbildungsbefehle erfordern würde, welche die individuelle Abbildungsvorrichtung haben kann. Ein Beispiel hierfür kann eine Vorrichtung des PostScript®-Typs sein, welche mit einem System verbunden ist, welches den Vorlagen-Beschreibungsdatenstrom wie er in dieser Erfindung genannt wird, überträgt und welche dann die Daten nach einer Übersetzung in die Post-Script-Sprache verarbeitet und druckt. Dies erfordert natürlich eine Einrichtung zum Stützen vorhandener Drucker.

Wie vorstehend ausgeführt, kann ein individueller Vorlagen-Beschreibungsdatenstrom entweder als ein Seitensatz oder als ein Bild definiert werden, wie in Fig. 1 dargestellt ist. Wenn eine Vorlage als ein Bild definiert ist, würde die Vorlage eine Seite oder eine Einseiten-Vorlage sein. Eine individuelle Vorlage kann aus einem einzigen Bild (Seite) oder aus einem einzigen Seitensatz gebildet sein. Ein Bild kreuzt jedoch nicht Seiten.

Die hierarchische Struktur eines Vorlagendatenstroms, wie er in Fig. 1 dargestellt ist, ist in Fig. 2 veranschaulicht. Seiten sind festgelegt bzw. definiert und gesetzt als der höchste Pegel in einer Bildhierarchie. Einheiten- bzw. Tokenfolgen (211) und (214) sind einzelne Elemente, welche Teile einer einzelnen Seite beschreiben können. Dies können beispielsweise eine Zeichnung, Text oder ein Bild sein. Andere Teile der Vorlage werden ebenfalls durch Tokenfolgen behandelt, welche Bildbeschreibungen erzeugen, welche verwendet werden, um einzelne Elemente in einem Bild festzulegen. Die Übersetzung von Bildern in eine Tokenfolge erlaubt eine wirksamere Verarbeitung des Vorlagendatenstroms und erlaubt auch ein schnelles Ändern von Elementen in einem Vorlagendatenstrom, wenn es notwendig ist oder gewünscht wird, und zwar viel leichter als es in herkömmlichen Systemen zulässig gewesen ist.

Insbesondere unter Bezugnahme auf Fig. 2 und 13 wird die "reine" Textverarbeitung dargestellt; jedoch hat selbstverständlich basierend auf der Token-Struktur und einer Stapelausführung, die nachstehend noch erläutert wird, ein Binärdatenstrom beispielsweise einen ersten Informationssatz in binärer Form hinsichtlich seiner Art, seines Werts und dann eine Längenkennung, wobei die Verarbeitung, die vorkommt, identisch ist.

Wie in Fig. 3 dargestellt, haben das Bild und der Datensatz eine festgesetzte "Grammatik", in welcher ein Bild entsprechend definiert ist, daß es einen wahlweisen Prolog und ein wahlweisen Bildkörper hat, wobei ein Seitensatz definiert ist, daß er einen wahlweisen Prolog und einen wahlweisen Seitensatz-Körper hat, und ein

Prolog Elemente hat, wie eine externe Deklaration, eine informative Deklaration, eine Ressourcen-Deklaration, eine Vorlagenerzeugungs-Befehls-Deklaration, eine Kontext-Deklaration, eine Wörterbuch-Erzeugungsdeklaration oder eine Einstellprozedur. Alle diese Elemente in dem Prolog sind wahlweise und können in dem Sinn wiederholt werden, daß sie vielfache Beispiele des selben Gegenstands sein können. Folglich können Vielfachpegel eines Prologs vorhanden sein, und das Handhaben dieser Vielfachbeispiele kann so, wie unten ausgeführt, ausgeführt werden.

Ein Bild-Körper besteht aus einem Bild oder einer Tokenfolge, welche entweder null- oder mehrmals vorkommen können, ebenso wie der Seitensatz-Körper, welcher einen Seitensatz oder Bilder hat, welche entweder null- oder mehrmals vorkommen können. In Fig. 3 ist veranschaulicht, daß ein Bild aus einem wahlweisen Prolog und wahlweisen Bild-Körpern besteht und daß ein Seitensatz aus wahlweisen Prologen und wahlweisen Seitensatz-Körpern besteht. Jeder Prolog besteht aus null oder mehr Prologelementen mit einem Bild-Körper, welcher aus null oder mehr Bildern oder Tokenfolgen besteht, wie bei 210 in Fig. 2 dargestellt ist. Wenn ein Bild in der Definition oder dem Datenstrom vorhanden ist, welcher den Bild-Körper bildet, wird dieses Bild auf einem Pegel unter dem Eltern-Bild in der hierarchischen Struktur untergebracht, wie in Fig. 2 dargestellt ist. Ein Seitensatz-Körper kann aus null oder mehr Seitensätzen bestehen, wie im Element 201 der Fig. 2 dargestellt ist.

Die generelle Architektur des Systems und der Datenstruktur ist in Fig. 4 dargestellt. Der Parser 40 erhält einen Bytestrom und kennzeichnet, verschiedene Einheiten (token). Bei der vorliegenden Erfindung sind die wichtigsten Einheiten, welche von dem Parser verarbeitet werden, der Bild-Beginn, das Bild-Ende, der Seitensatz-Beginn und das Seitensatz-Ende. Wenn diese Einheiten von dem Parser 400 erkannt werden, ruft der Parser eine entsprechende Routine auf, um die Verarbeitung des Datenstroms zu behandeln und eine Struktur einzustellen, wie sie beispielsweise in Fig. 13 dargestellt ist. Die herkömmliche Art von Parsing, was ohne weiteres entweder in Software oder in Hardware ausgeführt ist, wird an dieser Stelle verwendet. Folglich wird Parsing der Kürze halber hier nicht mehr erläutert.

Wenn beispielsweise ein "Beginn", wie in Fig. 13 dargestellt, angetroffen und von der Zentraleinheit (CPU) 1110 der Fig. 11 und insbesondere von dem Teil der Zentraleinheit 1110 erkannt wird, welche als ein Struktur-Prozessor wirkt, wird ein entsprechender Zeiger auf einen Bild-Seitensatz-Stapel (401) gerichtet. Die Funktionen des Struktur-Prozessors können in der Zentraleinheit 1110 oder in einem gesonderten, aber damit verbundenen Prozessor durchgeführt werden; der Einfachheit halber kann angenommen werden, daß der Struktur-Prozessor als ein Teil der Operationsfunktionen der Zentraleinheit 1110 ausgeführt ist. Wenn ein "Ende" erkannt wird, wird ein entsprechender Zeiger von dem Stapel weggenommen (popped), und der erforderliche Speicher wird gelöscht. Bei einem Klartext-Codieren einer Vorlage können beispielsweise Hinweissymbole (Bild) und (Seitensatz) verwendet, um einen Beginn anzuzeigen, während (<Bild>) und (<Seitensatz>) verwendet werden, um ein Ende zu bezeichnen. In einem binären Codierschema für den Vorlagendatenstrom können spezielle Hinweissymbole im Vergleich zu einem Bytezählern verwendet werden, um einen "Beginn"- und "En-

de"-Zustand für ein Bild und einen Seitensatz zu kennzeichnen. Folglich kann eine Stapelanordnung, bei welcher sogenannte Push- und Pop-Techniken eingesetzt werden, welche für die Verarbeitung von Datenfeldern bzw. Elementen in dem Stapel verwendet werden, benutzt werden, um den Beginn und das Ende eines vorgegebenen Seitensatzes oder Bildes festzulegen, welche das Handhaben und Kennzeichnen der "Anfänge" und "Enden" in einer sehr vorteilhaften Weise gestatten.

Wie in Fig. 5 dargestellt, enthält ein Zeiger zu einem Bild/ Seitensatz/Prolog eine Reihe von Zeigern, welche auf "null" oder auf den Speicher eingestellt sein können, welcher externen Deklarationen, informativen Deklarationen, Ressourcen-Definitionen, Ressourcen-Deklarationen, kontext-Deklarationen, einer Wörterbuch-Erzeugungsdeklaration, Einstellprozeduren oder Maschinenzuständen erforderlichenfalls zugeordnet ist. Wenn diese Datenfelder bzw. Elemente wahlweise vorgesehen sind und entweder null- oder mehrmals vorkommen können, wird in der vorliegenden Erfindung eine Zeigerranordnung verwendet, damit Mehrfachvorgänge oder Nullvorgänge ohne die daraus resultierende Speichervergeudung vorkommen können, was eintreten würde, wenn dies nicht der Fall wäre. Wenn Nullvorgänge der Fall sind, wird der Wert der Zeiger "null" und natürlich braucht kein Speicher diesen Vorgängen zugeordnet werden. Diese Zeiger sind mittels Bezugszeichen 503 bis 510 in Fig. 5 gekennzeichnet. Das Bezugszeichen 501 in Fig. 5 weist auf den Seitensatz-Pegel und das Bezugszeichen 502 weist auf den Bildpegel des einzelnen Bild/ Seitensatz-Prologes hin, was eine zweckmäßige und vorteilhafte Verarbeitung der einzelnen Bilder/Seitensätzen in der Struktur gestattet.

Diese Kennzeichnung, welche bezüglich jeder der Pegel 501 und 502 in Fig. 5 vorkommt, wird wichtig, wenn man bedenkt, daß eine einzige zu verarbeitende Seite viele Bestandteile haben kann, wie beispielsweise in Fig. 13 dargestellt ist. Die Bild-Pegel, die in Fig. 13 verarbeitet worden sind, ändern sich von einem oberen Pegel von eins auf einen unteren Pegel von zwei; jedoch kann selbstverständlich der Pegel auf einer komplizierteren Seite mit graphischen Darstellungen, welche andere graphische Darstellungen u.ä. enthalten, viele Bild-Pegel haben und der Pegel für Verarbeitungszwecke kann beträchtlich über zwei hinaus ansteigen. Mit Hilfe der Stapelanordnung werden die Grenzwerte bezüglich der Anzahl an Pegeln praktisch unbegrenzt und werden nur von dem Speicher 1120 begrenzt, welche in dem Stapel zur Verfügung steht, welcher von der Zentraleinheit (CPU) 1110 der Fig. 11 verwendet wird.

Wie in Fig. 6 dargestellt, wird eine verknüpfte List verwendet, um eine Spur der einzelnen Vorgänge oder Sachen in der in Fig. 5 dargestellten Struktur zu halten. Ein Zeiger beispielsweise zu einer in Fig. 5 dargestellten externen Deklaration 503 wird als ein Beispiel in Fig. 6 verwendet. Die Zeiger 504 bis 510 werden in ähnlicher Weise behandelt, wie der Zeiger 503. Alle Zeiger 503 bis 510 in Fig. 5 haben Seitensatz- und Bild-Pegel, die in der Datenstruktur enthalten sind, welche es gestattet, daß der Speicher, welcher diesen Vorgängen oder Sachen zugeteilt worden ist, freigegeben werden kann, wenn ein "Ende" für einen vorgegebenen Seitensatz oder ein vorgegebenes Bild erkannt wird. Dies erlaubt eine stärkere Nutzung von verfügbaren Speicher-Ressourcen durch das "Wiederbenutzen" des Speichers durch herkömmliche Speicher-Bereinigungs-Prozeduren. Ferner ermöglicht dies auch eine Art interner Fehlerüberprüfung, da eine fehlangepaßte Gruppe von "Anfängen" und "En-

den" ohne weiteres mittels der Zentraleinheit (CPU) 1110 der Fig. 11 und insbesondere in dem Abschnitt der Zentraleinheit 1110 festgestellt werden kann, welche als ein Strukturprozessor wirken kann. Diese Art Anordnung kann so sein, wie in den Ausführungsformen von Fig. 12A bis 12C dargestellt ist, in welchen die Verarbeitung in vielen verschiedenen Formen oder Umgebungen beispielsweise an dem Druckerende, dem Hauptprozessorende, d. h. am Vorlagendatenstrom-Creator oder an irgendeiner anderen Stelle dazwischen erreicht werden kann.

Wie in Fig. 6 dargestellt, zeigt der Zeiger 503 (Fig. 5) auf eine Datenstruktur einer Verknüpfungsliste. In der Struktur 600 ist 601 der Seitensatz-Pegel für die externe Deklaration. Der Bild-Pegel 602 ist ebenfalls in dieser Verknüpfungs-Struktur vorhanden. Ein externer Kennzeichner 603 und der Strukturtyp 604 sind ebenfalls vorgesehen. Der Zeiger auf externe Daten 605 gestattet das Speicherzuteilen zu den externen Daten selbst und 606 gestattet ein Verknüpfen von mehrfachen externen Deklarationen, indem ein Zeiger zu der nächsten Gruppe von Elementen vorgesehen wird, welche ähnlich dem Element 600 sein würden, aber für den nächsten Pegel "unten" in der Datenstruktur sein würden.

Das Verarbeiten, das für eine Bild-Beginn-Routine notwendig ist, ist in Fig. 7 und 13 dargestellt. Globale Variable, wie der Bild-Pegel (Pic-Pegel) stehen durch Verwenden einer der Routinen zur Verfügung, während der Bild-Pegel (501) und der Seitensatz-Pegel (502) lokale Variable sind, die in der Datenstruktur (500) enthalten sind, auf die durch die Eingabe in den Bild/Seitensatz-Stapel (401) in Fig. 4 gezeigt wird. Nach einem Starten beim Schritt 700 wird beim Schritt 701 der Pic-Pegel auf null bei der Initialisierung der gesamten Verarbeitung für einen Vorlagendatenstrom oder beim "Einschalten" initialisiert. Eine Fehlerüberprüfung erfolgt beim Schritt 702 und ein Fehler-Flag wird beim Schritt 703 gesetzt, wenn es zu einer Schwierigkeit kommt, worauf dann bezüglich einer weiteren Verarbeitung durch die Zentraleinheit 1110 (Fig. 11) beim Schritt 704 zurückgekehrt wird. Andernfalls wird eine Verarbeitung fortgesetzt und die globale Variable Pic-Pegel wird beim Schritt 705 inkrementiert. Wenn das Auswählen von Seiten notwendig wird, sollte der folgende Prozeß nach dem Schritt 705 eingefügt werden.

[Wenn (Pic-Pegel == 1), dann Seitenzahl ++]. Dies bedeutet, daß nach dem Schritt 705 der Pic-Pegel geprüft wird, um zu sehen, ob er eins ist. Wenn er eins ist, dann bezeichnet "Bildbeginn" das Beginnen des höchsten Pegels der Bildhierarchie, d. h. den Start einer Seite. Folglich wird die Seitenzahl inkrementiert. Die "Seitenzahl" wird initialisiert, so daß sie beim Einschalten einer Initialisierung oder bei einem Rücksetzen null wird. Die "Seitenzahl" sollte eine globale Variable sein, auf welche durch irgendeine Routine zugegriffen werden kann. Dieser Rücksprung ermöglicht die Fortsetzung einer Verarbeitung des Vorlagendatenstroms. Wenn der Pic-Pegel eine globale Variable ist, ist dieser Pegel über alle anderen Routinen verfügbar und kann für Mehrfach-Pegel durchgeführt werden. Eine Speicherzuteilung erfolgt beim Schritt 706 für die Bild/Seitensatz-Prolog-Datenstruktur (500) aus Fig. 5. Ein Zeiger auf die Datenstruktur wird beim Schritt 706 auf den Stapel (401) gerichtet. Der Bild-Pegel und der Seitensatz-Pegel der Datenstruktur (500) werden beim Schritt 707 gesetzt. Dann wird beim Schritt 708 bestimmt, ob die Eingabe die erste Eingabe des Stapsels ist. Wenn dem so ist, wird beim Schritt 710 ein Zeiger-Geräte-Zustand gesetzt, um auf

die Kopie der Ausgangs- oder Rückfallwerte für die Anzeige/Druckvorrichtung hinzuweisen, und die anderen Zeiger werden auf null gesetzt. Wenn es nicht der Fall ist, werden beim Schritt 709 alle Zeiger so gesetzt, daß sie auf dieselben Objekte wie der eine Eintrag zeigen, welcher unter diesem Eintrag in dem Stapel liegt. Dies erlaubt ein effizientes "Suchen" in dem Stapel durch den strukturierten Verarbeitungs-Routins/Ab schnitt der Zentraleinheit (CPU) 1110. An dieser Stelle sollte beachtet werden, daß die Strukturverarbeitung, die im Prozessor 1110 der Fig. 11 stattfindet, auch von einem separaten Prozessor durchgeführt werden kann, welcher in Verbindung mit der Zentraleinheit 1110 der Fig. 11 arbeitet, obwohl dies in einem generellen Schema nicht notwendig sein sollte.

Wie in Fig. 8 dargestellt, ist die Verarbeitung, welche für einen "Seitensatz-Beginn" erfolgt, ähnlich demjenigen, der in Fig. 7 dargestellt ist, außer daß die Fehlerüberprüfung auf dem PS-Pegel in 802 erfolgt, und es ist der PS-Pegel, welcher beim Schritt 805 inkrementiert wird. Sonst ist die Verarbeitung dieselbe für Seitensätze und "Bild", und sie wird der Einfachheit halber nicht weiter erläutert.

Anhand von Fig. 9 und 10 wird die Verarbeitung erläutert, welche als ein Ergebnis einer "Ende"-Routine vorkommt. Wie in Fig. 9 dargestellt, wird nach einem Start (Schritt 900) der Zustand des Stapels beim Schritt 901 mit dem laufenden Pic-Pegel geprüft, um die korrekte Verwendung der Bildstruktur in dem Vorlagendatenstrom zu gewährleisten und um irgendeine mögliche Korrumperung der Daten festzustellen und zu vermerken. Dies wird insbesondere wichtig, wenn der Vorlagendatenstrom nicht im Klartext vorliegt, sondern ein binärer Datenstrom ist, bei welchem die Länge des Datenstroms nicht bekannt ist oder infolge einer fehlerhaften Übertragung u.ä. als unrichtig bekannt ist. Eine Fehlerüberprüfung erfolgt auch beim Schritt 904 vor einem Weitergehen in der tatsächlichen Verarbeitung des "Bildende"-Befehls. Das Fehler-Flag wird beim Schritt 902 zusammen mit einem Rücksprung auf eine Verarbeitung gesetzt, was beim Schritt 903 erfolgt. Der obere Eintrag des Bild/Seitensatz-Stapels 401 wird beim Schritt 905 (aus dem Speicher) entnommen und anschließend wird der Speicher, welcher vorher verwendet wurde, für eine "Speicherbereinigung" freigegeben. Der diesbezügliche Speicher in der Bild/Seitensatz-Prolog-Datenstruktur, auf welche durch den entnommenen Eintrag hingewiesen worden ist, und die Verknüpfungsliste, auf welche durch die Zeiger in der Datenstruktur hingewiesen worden ist, haben einen Seitensatz-Pegel, welcher gleich dem PS-Pegel ist, und der Bild-Pegel, welcher gleich dem Pic-Pegel ist, wird erreicht, und der Pic-Pegel wird dann beim Schritt 609 um eins erniedrigt. Beim Schritt 907 wird dann festgestellt, ob der sich ergebende Pic-Pegel null ist, wenn dem so ist, ist beim Schritt 908 das Bild, welches gerade verarbeitet worden ist, das Bild mit dem höchsten Pegel (d. h. eine Seite), die Bilddaten können dann zum Drucken/Anzeigen in die Abgabeeinheit befördert werden, und der Bilddatenbereich wird zurückgesetzt, wodurch angezeigt wird, daß eine neue Seite zu verarbeiten ist. Wenn eine Auswahl von Seiten benötigt wird, sollte die "Seitenzahl" beim Schritt 908 geprüft werden, um zu bestimmen, ob die Seitenzahl unter den ausgewählten Seiten vorhanden ist.

Der Datenfluß in Fig. 10 ist ähnlich demjenigen, wie er in Fig. 9 dargestellt ist, außer daß sich die Bild- und Fehlerüberprüfung, wie sie in den Schritten 1001 und 1004 dargestellt ist, auf das Seitensatz-Ende bezieht, und

das Seitensatz-Pegel-Dekrementieren beim Schritt 1006 nicht das Drucken einer Seite oder ein Bildrücksetzen wie bei den Schritten 907 und 908 in der anhand von Fig. 9 erläuterten Bildende-Routine ist.

#### Patentansprüche

1. Einrichtung zum Steuern der Darstellung einer Vorlage mittels einer Darstellungsvorrichtung, gekennzeichnet durch

eine Eingabeeinrichtung zum Aufnehmen eines eingegebenen Vorlagendatenstroms, welcher die Vorlage festlegt, wobei der Datenstrom aus einem Bild oder einem Seitensatz mit einer hierarchisch geordneten Struktur zusammengesetzt ist, wobei das Bild null oder mehr Prolog und null oder mehr Bildkörper aufweist, und der Seitensatz null oder mehr Prolog und null oder mehr Seitensatzkörper aufweist,

wobei der Bildkörper null oder mehr Tokenfolgen und null oder mehr Bilder aufweist, wobei der Seitensatz-Körper null oder mehr Bilder und null oder mehr Seitensätze aufweist;

eine Strukturverarbeitungseinrichtung, die mit der Eingabeeinrichtung verbunden ist, um Seitensatz-Definitionen oder Bild-Definitionen festzulegen, um einen Anfang und ein zugeordnetes Ende jeder Seitensatz- oder Bilddefinition festzustellen, wobei die Strukturverarbeitungseinrichtung jeden Prolog in dem festgelegten Seitensatz oder den festgelegten Bild-Definitionen verarbeitet, Körperteile zwischen jedem festgestellten Anfang und dem zugeordneten festgestellten Ende in einer geordneten Hierarchie ordnet und abgegebene Vorlageninstruktionen zum Steuern der Darstellungsvorrichtung erzeugt und abgibt, und

eine Abgabeeinrichtung, welche mit der Strukturverarbeitungseinrichtung verbunden ist, um die abgegebenen Vorlageninstruktionen an die Darstellungsvorrichtung anzulegen.

2. Einrichtung zum Steuern der Darstellung einer Vorlage durch eine Darstellungsvorrichtung, gekennzeichnet durch

eine Eingabeeinrichtung zum Aufnehmen eines eingegebenen Vorlagendatenstroms, welcher die Vorlage festlegt, wobei der Datenstrom aus einem hierarchisch geordneten Satz von einem oder mehreren Prologen zusammengesetzt ist, die jeweils null oder mehr zugeordnete Inhaltsteile haben;

eine Strukturverarbeitungseinrichtung, welche mit der Eingabeeinrichtung verbunden ist, um jeden Prolog zu verarbeiten, um Seitensatz-Definitionen und Bildelemente des Vorlagendatenstroms festzulegen, wobei die Strukturverarbeitungseinrichtung die festgelegten Seitensatz-Definitionen verarbeitet, um einen Beginn und ein zugeordnetes Ende jeder Seitensatz-Definition festzustellen, Inhaltsteile zwischen jedem festgestellten Beginn und einem zugeordneten festgestellten Ende in einer geordneten Hierarchie ordnet, und abgegebene Vorlageninstruktionen zum Steuern der Darstellungseinrichtung erzeugt und abgibt, und

eine Abgabeeinrichtung, welche mit der Strukturverarbeitungseinrichtung verbunden ist, um die abgegebenen Vorlageninstruktionen an die Darstellungseinrichtung anzulegen.

3. Einrichtung nach Anspruch 2, gekennzeichnet durch eine Einrichtung zum Steuern eines Bereichs

in der Ordnungshierarchie eines gegebenen Seiten-  
satzes und Bildes.

4. Einrichtung nach Anspruch 2, dadurch gekenn-  
zeichnet, daß die Steuereinrichtung einen Stapel-  
speicher aufweist. 5

5. Einrichtung nach Anspruch 2, gekennzeichnet  
durch eine Speichereinrichtung, welche mit der  
Strukturverarbeitungseinrichtung verbunden ist,  
um die null oder mehr zugeordneten Inhaltsteile zu  
speichern, welche durch einen verarbeiteten Prolog 10  
festgelegt sind.

6. Einrichtung nach Anspruch 2, gekennzeichnet  
durch eine Einrichtung zum Speicher und Darstel-  
len einer Seitenzahl.

7. Einrichtung nach Anspruch 2, gekennzeichnet 15  
durch eine Einrichtung, um selektiv Seiten aus dem  
eingegebenen Vorlagenseitenstrom darzustellen.

8. Einrichtung nach Anspruch 5, gekennzeichnet  
durch einen syntaktischen Analyseabschnitt zum  
Prüfen einer syntaktischen Richtigkeit jedes festge-  
stellten Beginns und jedes zugeordneten Endes je- 20  
der Seitensatz-Definition.

9. Einrichtung nach Anspruch 5, dadurch gekenn-  
zeichnet, daß die Strukturverarbeitungseinrichtung  
einen Staplespeicher aufweist, um eine hierarchi- 25  
sche Ordnung der null oder mehr Inhaltsteile zu  
steuern, welche von der Speichereinrichtung ge-  
speichert sind.

10. Einrichtung nach Anspruch 9, dadurch gekenn-  
zeichnet, daß der Staplespeicher eine Zeigerstruk- 30  
tur benutzt, um eine Kennzeichnung der gespei-  
cherten Elemente zu erhalten.

11. Einrichtung nach Anspruch 9, dadurch gekenn-  
zeichnet, daß die Staplespeichereinheit einen Sta- 35  
pelspeicher aufweist, welcher eine verknüpfte Liste  
von Zeigern speichert, welche die geordnete Hier-  
archie kennzeichnen und dadurch einen Suchweg  
schaffen, welcher durch die verknüpfte Liste von  
gespeicherten Zeigern festgelegt ist.

12. Einrichtung nach Anspruch 9, gekennzeichnet 40  
durch eine Einrichtung, um für ein Wiederbenutzen  
gespeicherte Zeiger in der geordneten Hierarchie  
freizugeben, welche den Inhaltsteilen des Vorla-  
gendaströms entsprechen, die auf dem niedrigen  
Pegel der Hierarchie sind, relativ zu den Inhaltstei- 45  
len der dargestellten Seite in der geordneten Hier-  
archie nach einer Darstellung einer durch die Sei-  
tensatz-Definition festgelegten Seite und der Dar-  
stellung aller Seiten in dem Vorlagendaström,  
welche auf einem niedrigeren Pegel in der geordne- 50  
ten Hierarchie sind.

13. Einrichtung zum Steuern der Darstellung einer  
Vorlage mittels einer Darstellungsvorrichtung ge-  
kennzeichnet durch  
eine Eingabeeinrichtung zum Aufnehmen eines ein-  
gegebenen Vorlagendaströms, welcher die Vor- 55  
lage festlegt, wobei der Datenstrom zumindest ei-  
nen Prolog und null oder mehr zugeordnete In-  
haltsteile aufweist;  
eine CPU-Einheit, welche mit der Eingabeeinrich-  
tung verbunden ist, um den Datenstrom zu verar-  
beiten und die abgegebenen Vorlageninstruktionen  
zu erzeugen, die auf dem verarbeiteten Datenstrom  
basieren, mit  
einer Einrichtung, um ein Adressieren zumindest 60  
des einen Prologs und der zugeordneten Inhaltsteile  
in einer geordneten Hierarchie zu ordnen, und mit

einer Einrichtung zum Verarbeiten zumindest eines  
Prologs und zugeordneter Inhaltsteile in der geord-  
neten Hierarchie, um die abgegebenen Vorlagenin-  
struktionen zu erzeugen, und eine Abgabeeinrich-  
tung, welche mit der CPU-Einheit verbunden ist,  
um die abgegebenen Vorlageninstruktionen an die  
Darstellungsvorrichtung für die Darstellung der  
Vorlage anzulegen, welche durch die abgegebenen  
Vorlageninstruktionen dargestellt ist.

14. Einrichtung nach Anspruch 13, dadurch gekenn-  
zeichnet, daß die Ordnungseinrichtung eine Stapele-  
speichereinheit aufweist, um Zeiger zu speichern,  
die Speicherstellen zum Speichern von Prologen  
und zugeordneten Inhaltsteilen in einer wieder auf-  
zufindenden Folge zu ordnen, welche durch die  
Zeiger festgelegt ist.

15. Verfahren zum Steuern der Darstellung einer  
Vorlage, welche durch einen Vorlagendaström  
festgelegt ist, dadurch gekennzeichnet, daß  
a) in eine Prozessoreinheit ein Vorlagendaström  
eingegeben wird, welcher eine darzu-  
stellende Vorlage festlegt, wobei der Daten-  
strom Seitensatz-Definitionen einschließt, wo-  
bei jede Seitensatz-Definition einen Beginn,  
null oder mehr Bildelemente und ein Ende ein-  
schließt, die Bildelemente null oder mehr Pro-  
logen aufweisen, jeder Prolog null oder mehr  
Prologelemente und Inhaltsteile aufweist und  
die Inhaltsteile bilddefinierende Tokenfolge-  
Elemente enthalten;

b) der eingegebene Bilddatenstrom eine auto-  
matische Syntaxanalyse (Parsing) erfährt, um  
den Beginn und das Ende eines Prologs in dem  
eingegebenen Vorlagendaström festzule-  
gen, um dessen Existenz zu bestimmen;

c) ein Speicher bei einem gegebenen hierarchi-  
schen Speicherordnungspegel zugeordnet  
wird, in dem zugeordnetem Speicher beiden  
gegebenen hierarchischen Pegel Vorlagenda-  
ten geordnet und gespeichert werden, welche  
zwischen einem gegebenen, festgestellten Be-  
ginn und einem zugeordneten festgestellten  
Ende eines Prologs definiert sind;

d) auf einen Stapel ein Zeiger zeigt, welcher  
einen Speicher kennzeichnet, der bei dem ge-  
gebenen hierarchischen Pegel zugeordnet ist;

e) ein Speicher mit einem niedrigeren hierar-  
chischen Pegel unter dem gegebenen hierar-  
chischen Pegel zugeordnet wird und an dem  
niedrigeren hierarchischen Pegel Vorlagenda-  
ten gespeichert werden, welche zwischen ei-  
nem weiteren festgestellten Beginn, welcher  
vor dem festgestellten Ende, das dem gege-  
benen festgestellten Beginn zugeordnet ist, fest-  
gestellt wird und einem festgestellten Ende de-  
finiert ist, das dem weiteren festgestellten Be-  
ginn zugeordnet ist, und

f) auf den Stapel ein Zeiger zeigt, welcher ei-  
nen Speicher kennzeichnet, welcher bei dem  
niedrigeren hierarchischen Pegel zugeordnet  
ist.

16. Verfahren nach Anspruch 15, dadurch gekenn-  
zeichnet, daß aus dem Stapel Zeiger herausgeholt  
werden (popping), welche den Inhaltsteilen des  
Vorlagendaströms entsprechen, welche auf ei-  
nem niedrigeren hierarchischen Pegel bezüglich  
der Inhaltsteile der dargestellten Seite bei der Dar-  
stellung einer Seite, welche durch die Seitensatz-

Definition festgelegt ist, und bei der Darstellung aller Seiten in der Vorlage sind.

17. Verfahren nach Anspruch 15, dadurch gekennzeichnet, daß die Schritte (e) und (f) wiederholt werden, wenn ein Beginn mit einem niedrigeren hierarchischen Pegel vor einem Feststellen eines Endes festgestellt wird, das einem festgestellten Beginn mit einem höheren hierarchischen Pegel zugeordnet ist.

18. Prozessor zum Steuern einer Darstellung einer Vorlage, welche durch einen Vorlagendatenstrom festgelegt ist, wobei der Prozessor programmiert ist, um die Schritte durchzuführen, daß

a) in eine Prozessoreinheit ein Vorlagendatenstrom eingegeben wird, welcher eine darzustellende Vorlage festlegt, wobei der Vorlagendatenstrom Seitensatz-Definitionen einschließt, wobei jede Seitensatzdefinition einen Beginn > null oder mehr Bildelemente und ein Ende einschließt, die Bildelemente null oder mehr Prolog aufweisen, jeder Prolog null oder mehr Prologelemente und Inhaltsteile aufweist, und die Inhaltsteile bilddefinierende Tokenfolge-Elemente enthalten;

b) der eingegebene Vorlagendatenstrom einer automatischen Syntaxanalyse unterzogen wird (parsing), um den Beginn und das Ende eines Prologs in dem eingegebenen Vorlagendatenstrom festzulegen, um dessen Existenz zu bestimmen;

c) ein Speicher mit einem vorgegebenen hierarchischen Speicherzuordnungspegel zugeordnet wird, in dem zugeordneten Speicher bei dem gegebenen hierarchischen Pegel Vorlagendaten geordnet und gespeichert werden, welche zwischen einem gegebenen, festgestellten Beginn und einem zugeordneten, festgestellten Ende eines Prologs festgelegt worden sind;

d) auf einen Stapel ein Zeiger weist, der einen Speicher kennzeichnet, der bei dem gegebenen hierarchischen Pegel zugeordnet ist,

e) ein Speicher mit einem niedrigeren hierarchischen Pegel unter dem vorgegebenen hierarchischen Pegel zugeordnet wird und bei dem niedrigeren hierarchischen Pegel Vorlagendaten gespeichert werden, welche zwischen einem weiteren festgestellten Beginn, welcher vor dem festgestellten Ende festgestellt worden ist, das dem gegebenen festgestellten Beginn zugeordnet ist, und einem festgestellten Ende festgelegt sind, das dem weiteren festgestellten Beginn zugeordnet ist, und

f) auf den Stapel ein Zeiger weist, welcher einen Speicher kennzeichnet, der bei dem niedrigeren hierarchischen Pegel zugeordnet worden ist.

19. Computer-lesbares Medium zum Speichern eines Programms zum Steuern einer Darstellung einer Vorlage, welche durch einen Vorlagendatenstrom festgelegt ist, wobei das Programm die Schritte definiert, daß

a) in eine Prozessoreinheit ein Vorlagendatenstrom eingegeben wird, welcher eine darzustellende Vorlage festlegt, wobei der Vorlagendatenstrom Seitensatz-Definitionen einschließt, wobei jede Seitensatzdefinition einen Beginn, null oder mehr Bildelemente und ein

Ende einschließt, die Bildelemente null oder mehr Prolog aufweisen, jeder Prolog null oder mehr Prologelemente und Inhaltsteile aufweist, und die Inhaltsteile Bilddefinierende-Tokenfolge-Elemente enthalten;

b) der eingegebene Vorlagendatenstrom einer automatischen Syntaxanalyse unterzogen wird (parsing), um den Beginn und das Ende eines Prologs in dem eingegebenen Vorlagendatenstrom festzulegen, um dessen Existenz zu bestimmen;

c) ein Speicher mit einem vorgegebenen hierarchischen Speicherzuordnungspegel zugeordnet wird, in dem zugeordneten Speicher bei dem gegebenen hierarchischen Pegel Vorlagendaten geordnet und gespeichert werden, welche zwischen einem gegebenen, festgestellten Beginn und einem zugeordneten festgestellten Ende eines Prologs festgelegt sind;

d) auf einen Stapel ein Zeiger hinweist, welcher einen Speicher kennzeichnet, welcher bei dem gegebenen hierarchischen Pegel zugeordnet ist;

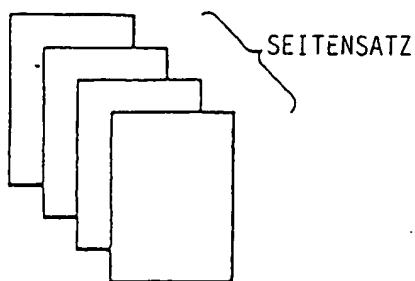
e) ein Speicher mit einem niedrigeren hierarchischen Pegel unter dem vorgegebenen hierarchischen Pegel zugeordnet wird, und bei dem niedrigeren hierarchischen Pegel Vorlagendaten gespeichert werden, welche zwischen einem weiteren festgestellten Beginn, welcher vor dem festgestellten Ende festgestellt wird, das dem vorgegebenen festgestellten Beginn zugeordnet ist, und einem festgestellten Ende festgelegt sind, das dem weiteren festgestellten Ende zugeordnet ist,

f) auf den Stapel ein Zeiger weist, welcher einen Speicher kennzeichnet, der bei dem niedrigeren hierarchischen Pegel zugeordnet ist.

Hierzu 13 Seite(n) Zeichnungen

**- Leerseite -**

VORLAGE = SEITENSATZ



VORLAGE = EINE SEITE

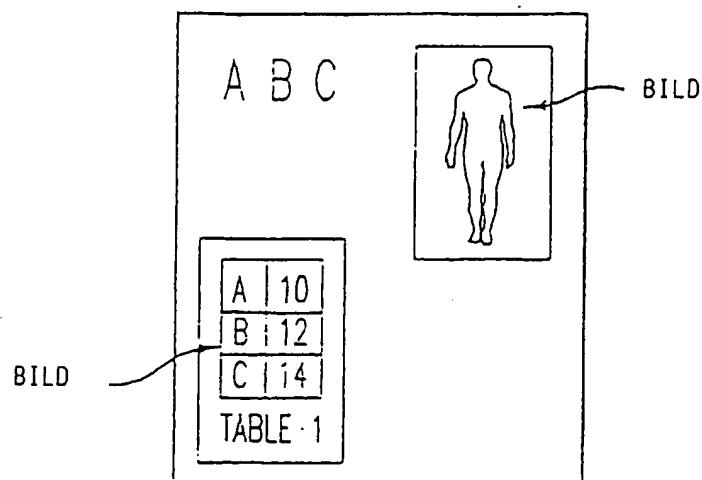


FIG. 1

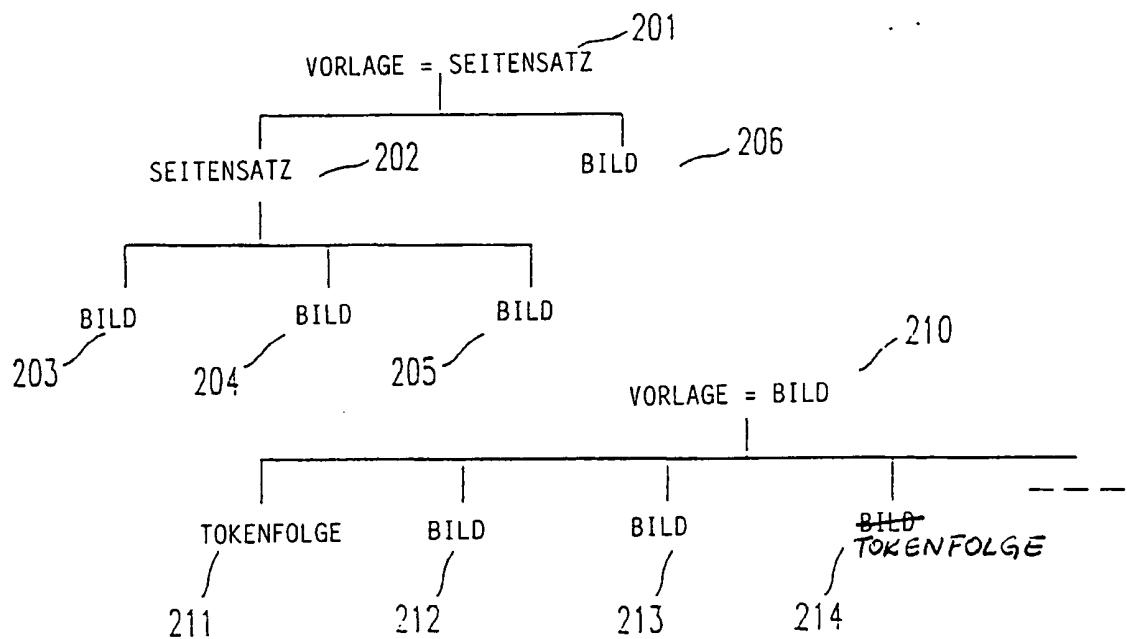


FIG. 2

```

BILD      :: = (PROLOG)?, (BILD-KÖRPER)?
SEITENSATZ :: = (PROLOG)?, (SEITENSATZ-KÖRPER)?
PROLOG    :: = (EXTERNE DEKL.*,
                INFORMATIVE DEKL.*  

                RESSOURCEN-DEF.*,  

                RESSOURCEN-DEKL.*  

                DOK.-PROD.-INSTR.-DEKL.*,  

                KONTEXT DEKL.*,  

                WÖRT-ERZEU.G.-DEKL.*  

                EINSTELL-PROZEDUR*)
BILD-KÖRPER (BILD | TOKENFOLGE)*
SEITENSATZ-KÖRPER = (SEITENSATZ | BILD)*
  
```

- , ALLE MÜSSEN IN DARGESTELLTER REIHENFOLGE VORKOMMEN
- | EINE UND NUR EINE MUSS VORKOMMEN
- ? WAHLWEISE (0 ODER 1-MAL)
- \* WAHLWEISE UND WIEDERHOLBAR (0 ODER MEHRMALS)

*FIG. 3*

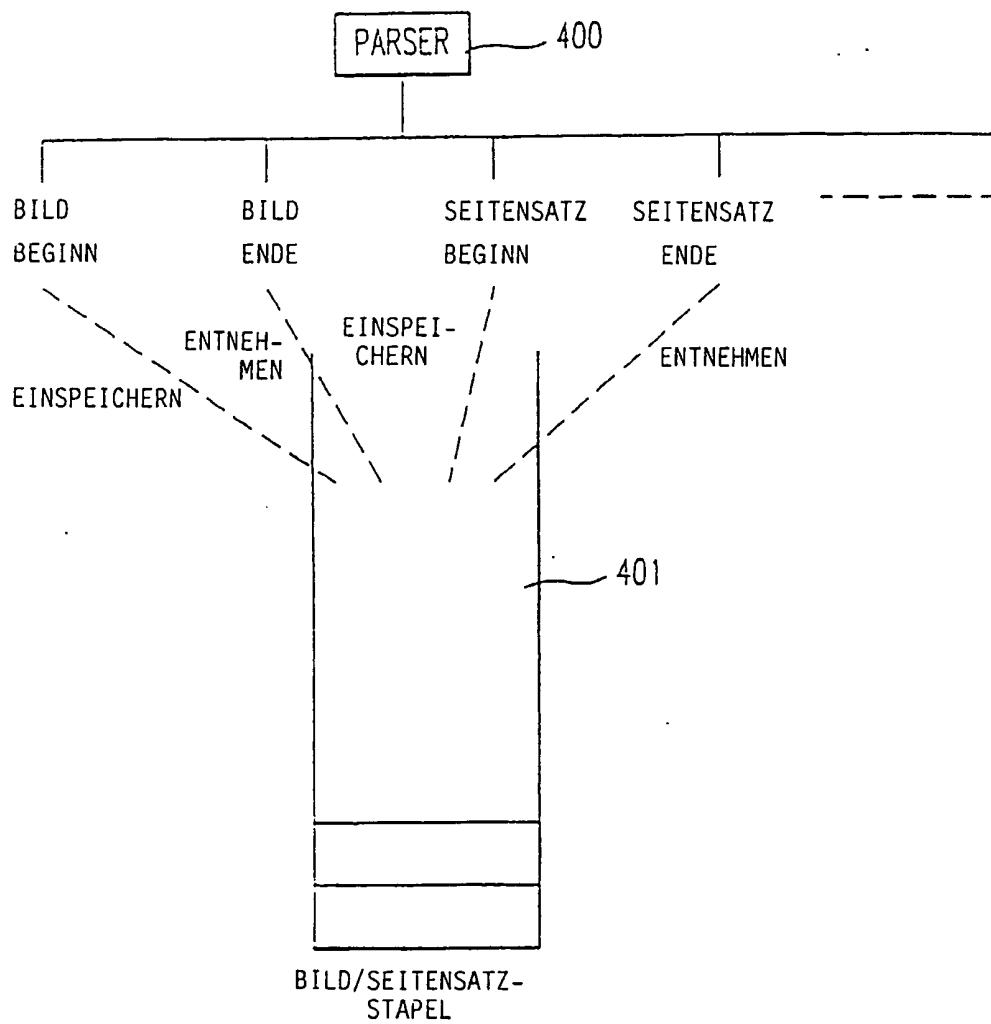


FIG. 4

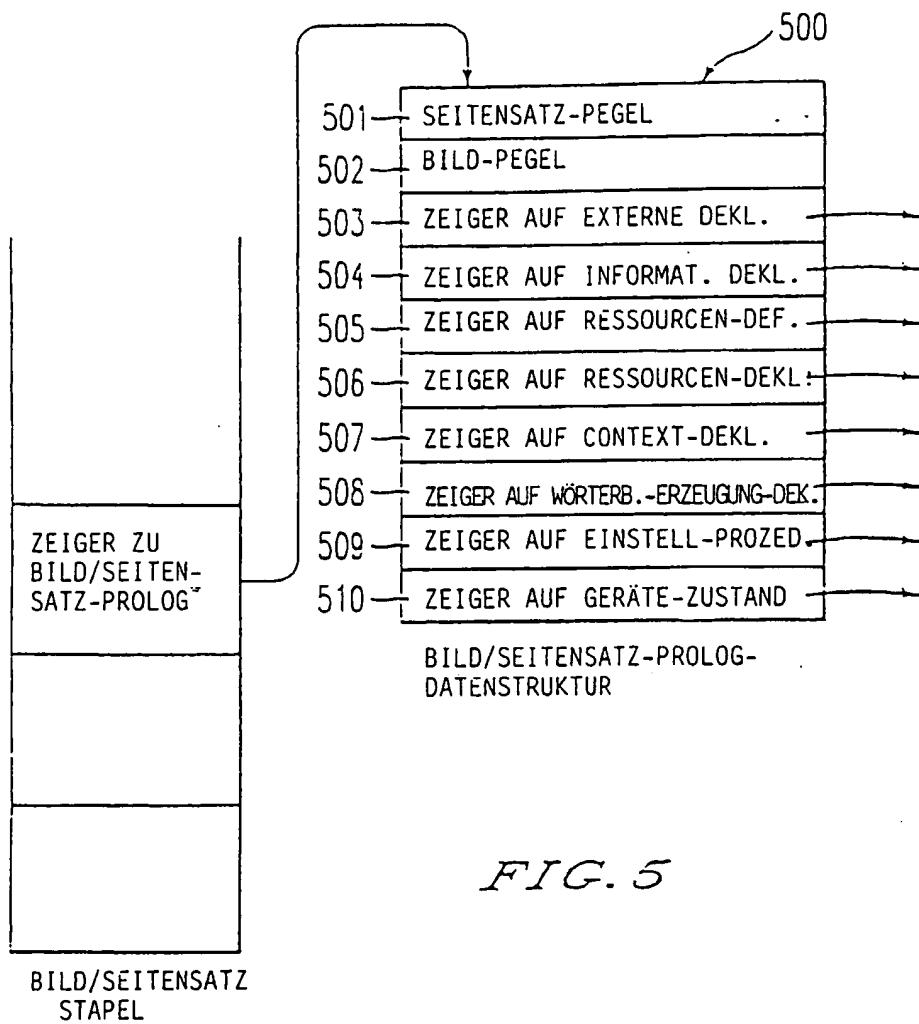


FIG. 5

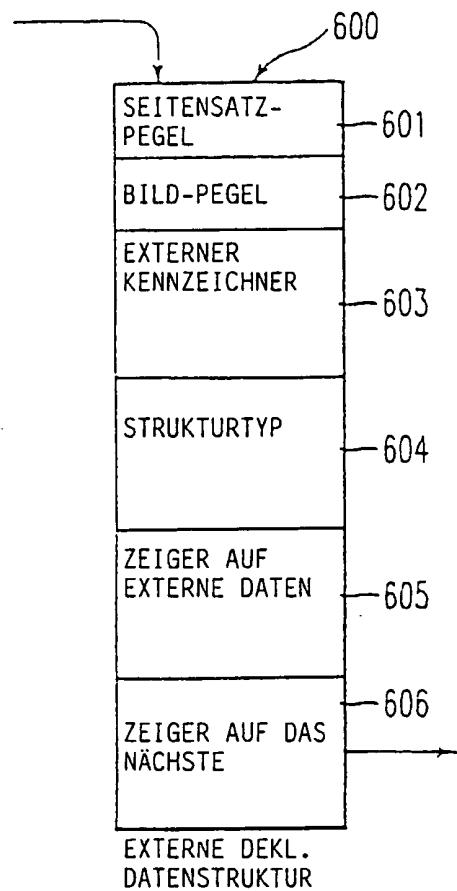
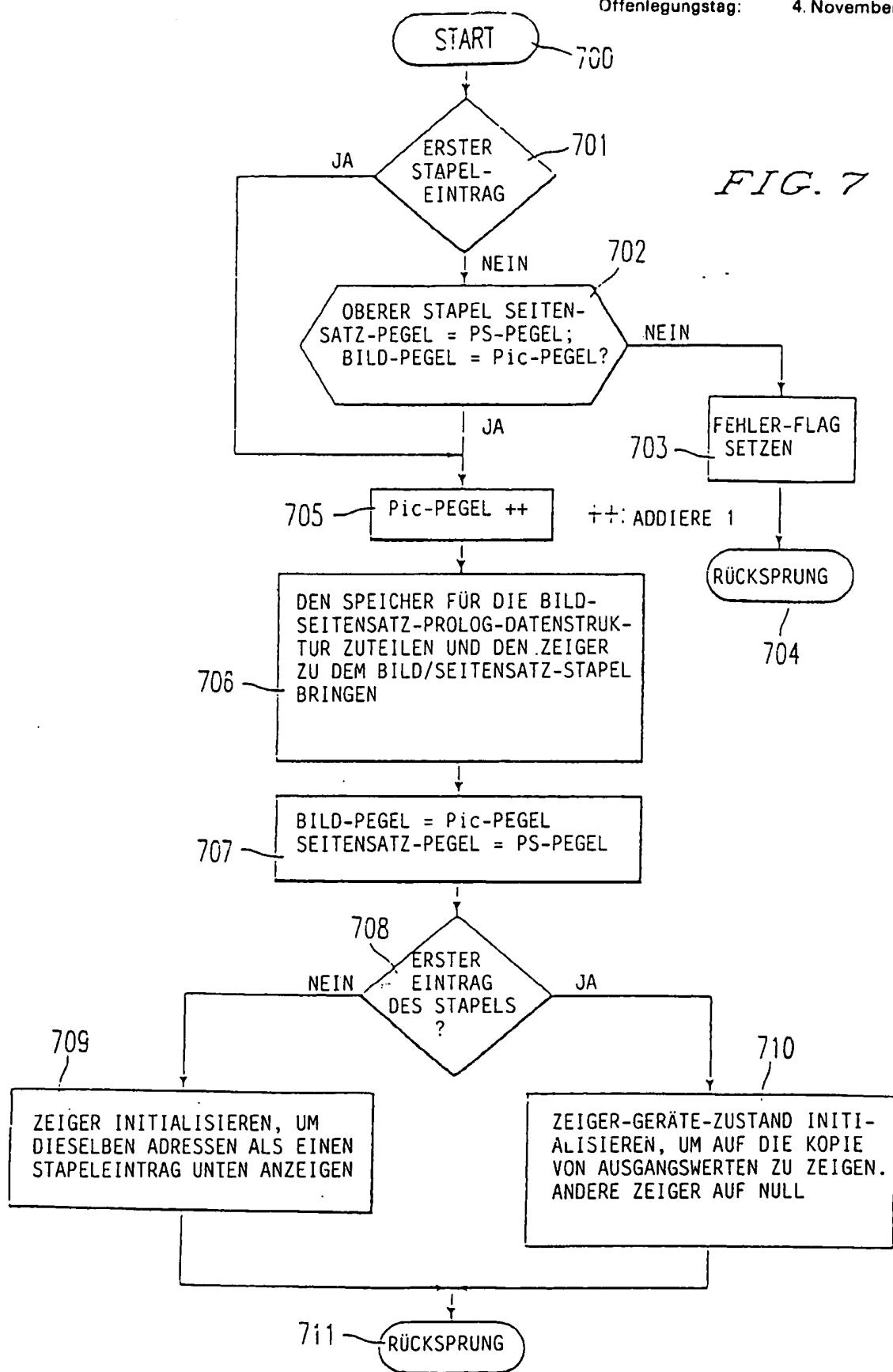
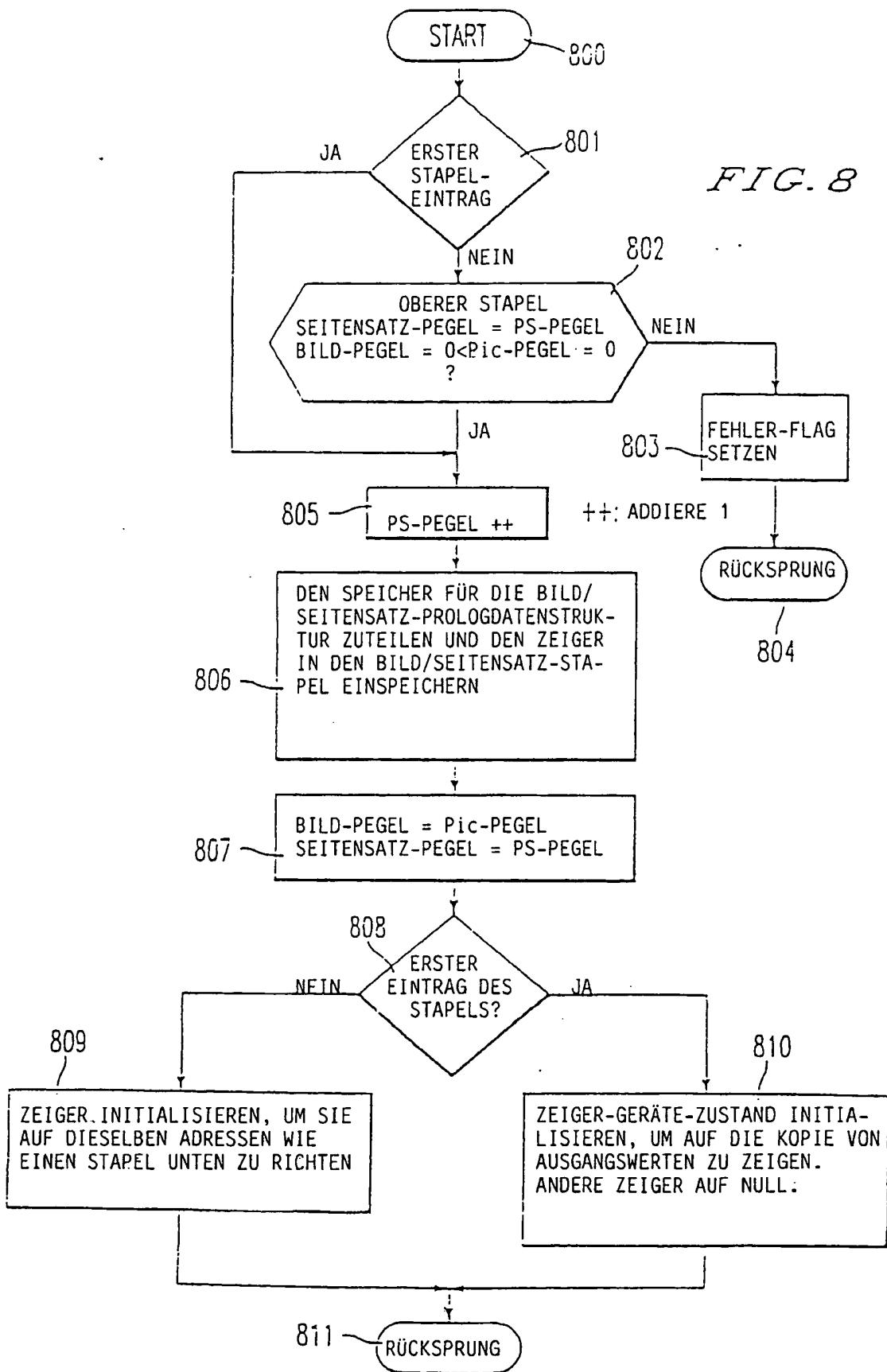
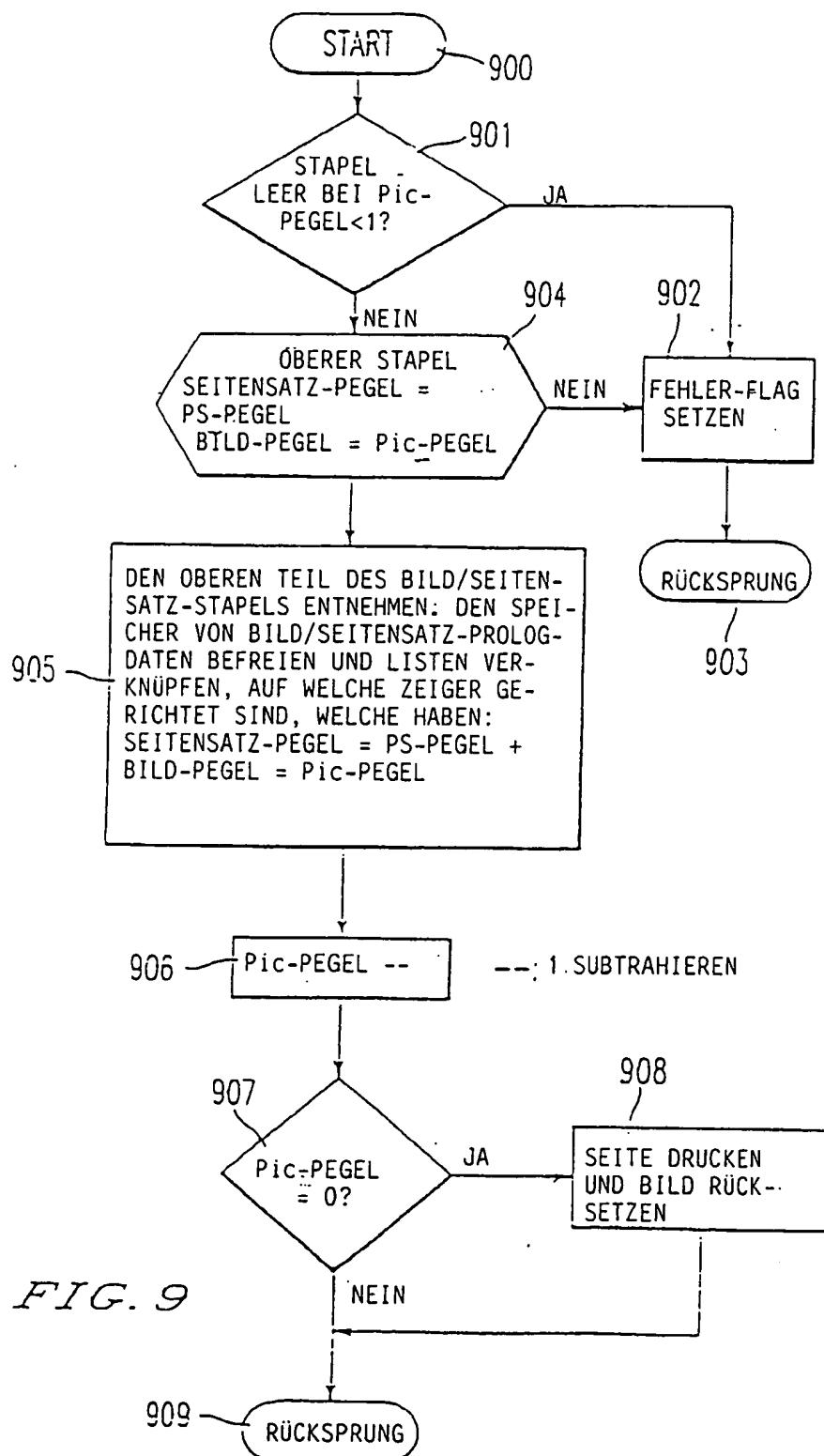


FIG. 6







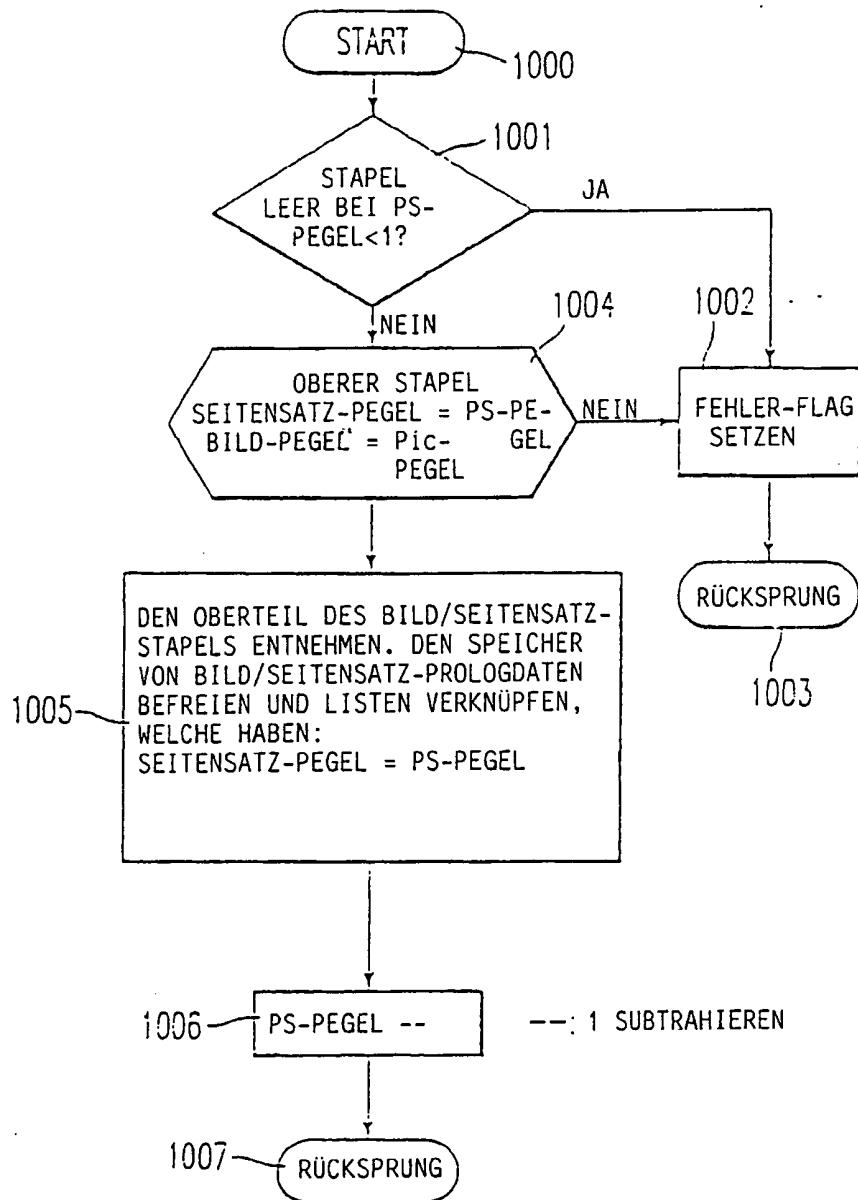


FIG. 10

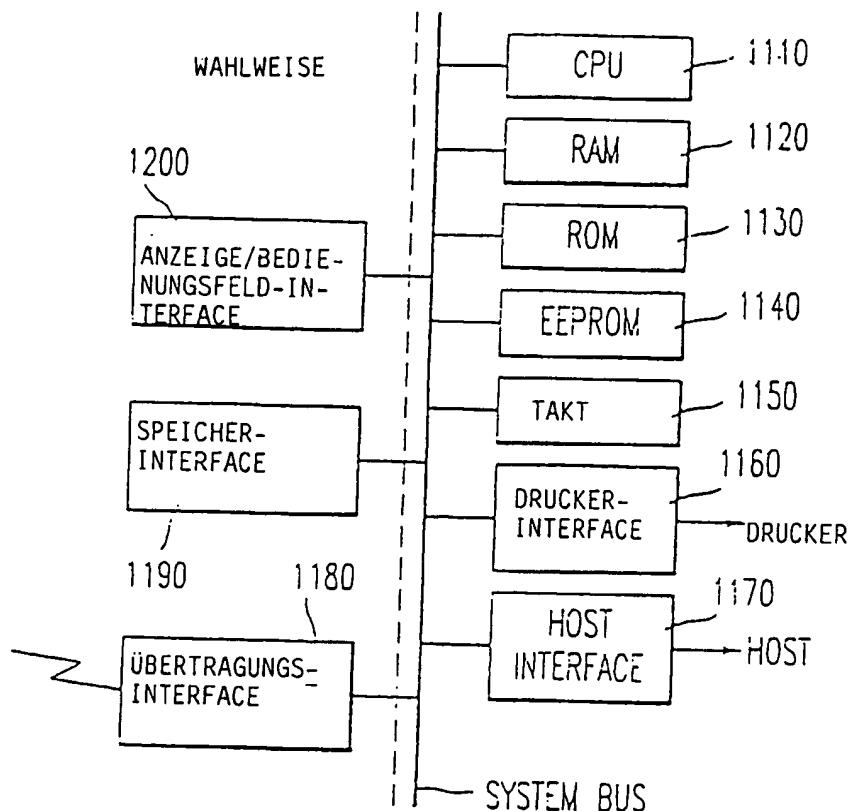


FIG. 11

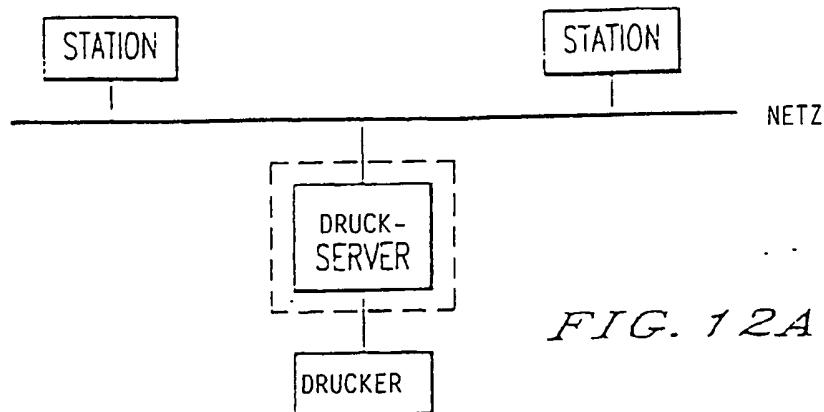


FIG. 12A

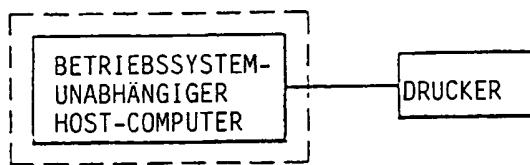


FIG. 12B

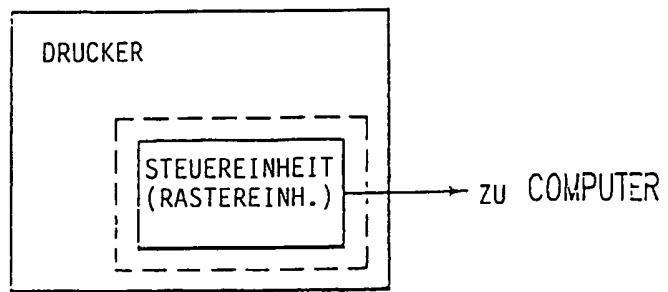


FIG. 12C

